

Turtle Graphics

Let's see what we can draw on Python!

A first Object: Logo Turtle

- Dr. Seymour Papert at MIT invented the Turtle as a graphical and mathematical object to think with for the children's programming language, Logo (1966)

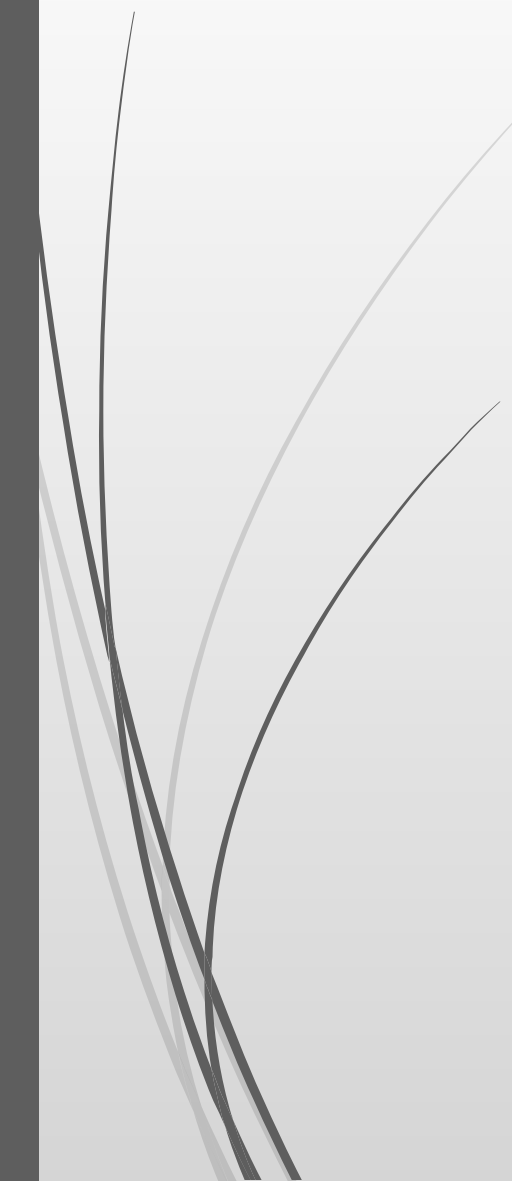


- Children programmed robot turtles to draw pictures





Turtle Graphics

- ❑ Turtle uses the foundation of a coordinate (xy) plane in order to help draw its graphics
 - ❑ Python has a built-in module that supports turtle graphics called “turtle”
 - ❑ Importing this module gives you access to all the functions necessary in order to draw vectors on the screen
- 



Turtle Graphics

- In Turtle, you control a cursor, also known as the “turtle”
- It has the following properties:
 1. A position in 2D space on the coordinate plane
 2. An orientation, or heading
 3. A pen that can lay down color on the canvas



Setting up your environment

#first we need to import the turtle module

```
import turtle
```

```
timmy = turtle.Turtle() # declaring a variable named timmy  
                        # of type turtle.
```

set a title for your window

```
timmy.title("My turtle animation")
```

set up the screen size (in pixels – 400 x 400)

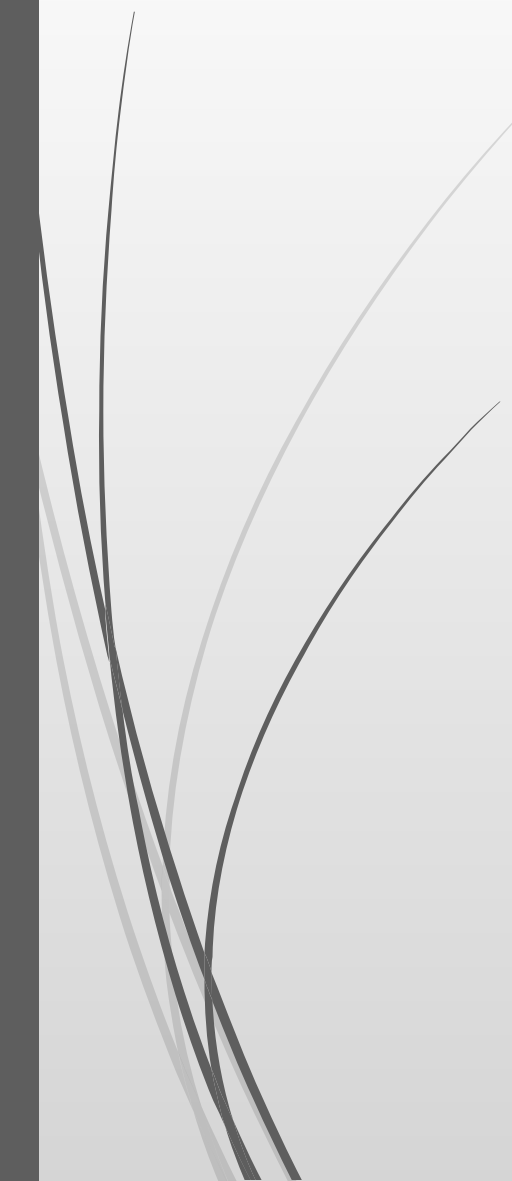
set up the starting point of the turtle (0,0) – origin

```
timmy.setup(400,400,0,0)
```



Setting up your environment

<https://hourofpython.trinket.io/a-visual-introduction-to-python#/turtles/meet-tina>

- However this website was somewhat created for the sake of using Turtle, not so much Python in it's entirety
 - So, the environment is already set up for you as a 400 x 400 canvas window and the turtle is set to start at the origin (0, 0)
- 

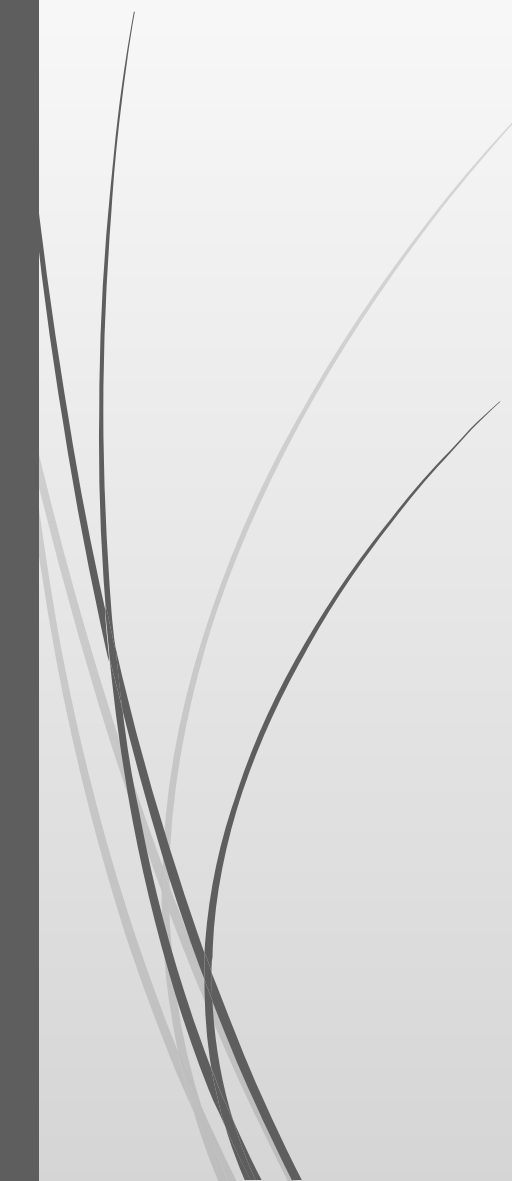
trinket.io/Python

The screenshot shows the trinket.io web application in a browser. The address bar displays the URL `https://trinket.io/library/trinkets/create?lang=python`. The browser's bookmark bar includes links to 'Apps', 'Bookmarks', 'Genesis', 'AP Calc AB', 'CSCI-UA.0002 : Fall ...', 'River Dell Mail', 'DyKnow', and 'The Team - RDHS C...'. The trinket.io header features the logo, a 'Home' link, and navigation options for 'Hour of Python', 'Help', a user profile 'donaldseok', and a 'Sign Out' button. The main interface has a breadcrumb trail: 'Home / My Trinkets / Untitled'. On the right side of this trail are 'Save' and 'Cancel' buttons. Below the breadcrumb is a toolbar with a hamburger menu, the trinket logo, a 'Run' button, and a '? Modules' link. The central area is a code editor with a file tab labeled 'main.py'. The editor shows a single line of code at line 1, which is currently blank. A vertical scrollbar is visible on the right side of the code editor. In the bottom right corner of the page, there is a blue circular chat button with a white question mark icon.

This is a Python trinket. Type or paste Python code you want your students to start with in the box on the left. Click ► **Run** to view its output in the box



Important Note!

- ❑ Do not name your Python source code file “turtle.py”
 - ❑ This will prevent Python from finding the correct “turtle.py” module when using the import turtle statement at the top of your program
- 



Getting rid of your turtle window

- ❑ This function call will cause your turtle window to deactivate when you click on it. Place it at the end of your turtle program.

```
turtle.exitonclick()
```

- ❑ This will not be applicable on the website



Basic Turtle Functions

- ❑ You can move your turtle forward by using the following command:

`turtle.forward(pixels)` #pixels can also be denoted by the number of spaces you want to move

- ❑ And you can have your turtle turn by using the following commands:

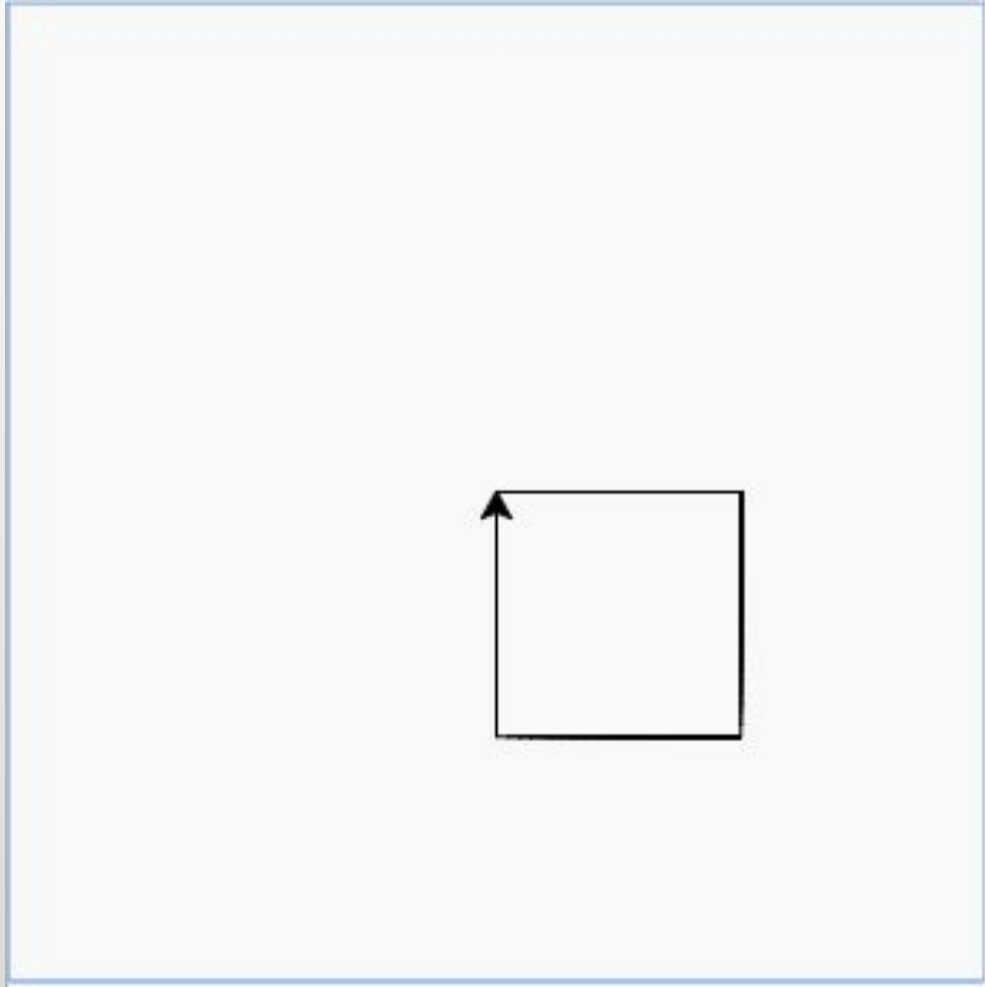
`turtle.right(degrees)`

`turtle.left(degrees)`

- ❑ Your turtle will continually “draw” while it’s moving

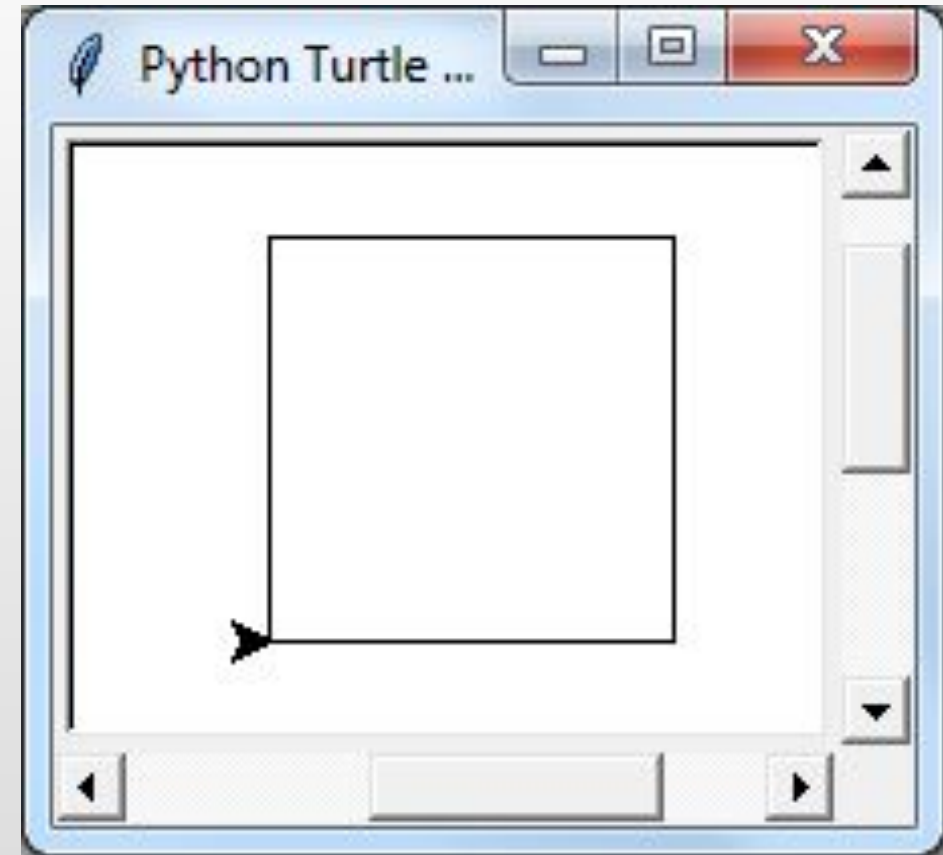
Programming Exercise

- Draw a box

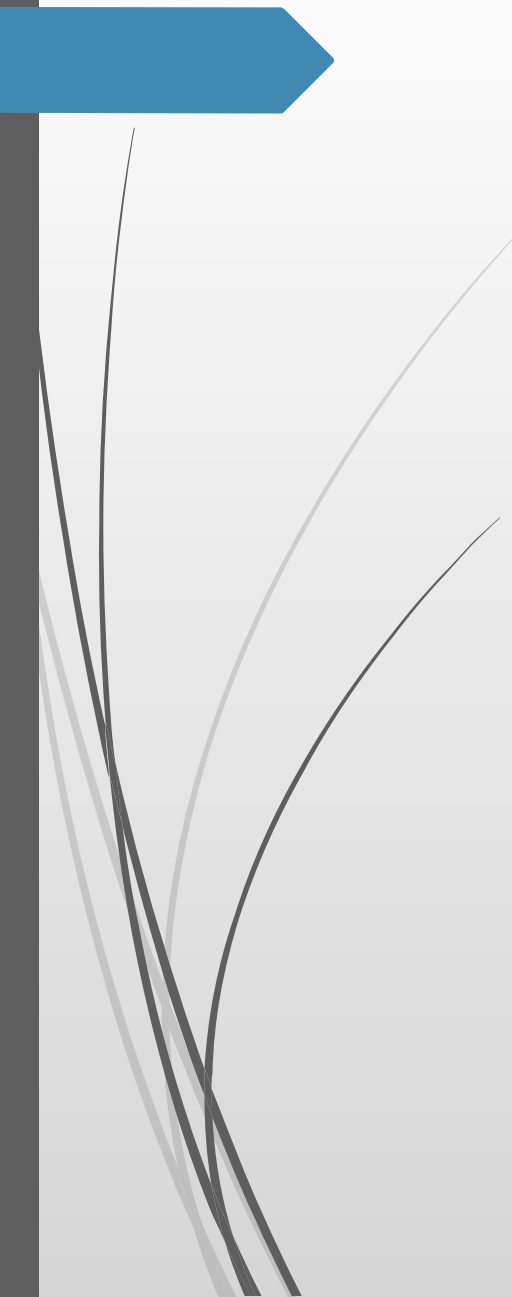


File Edit Format Run Options

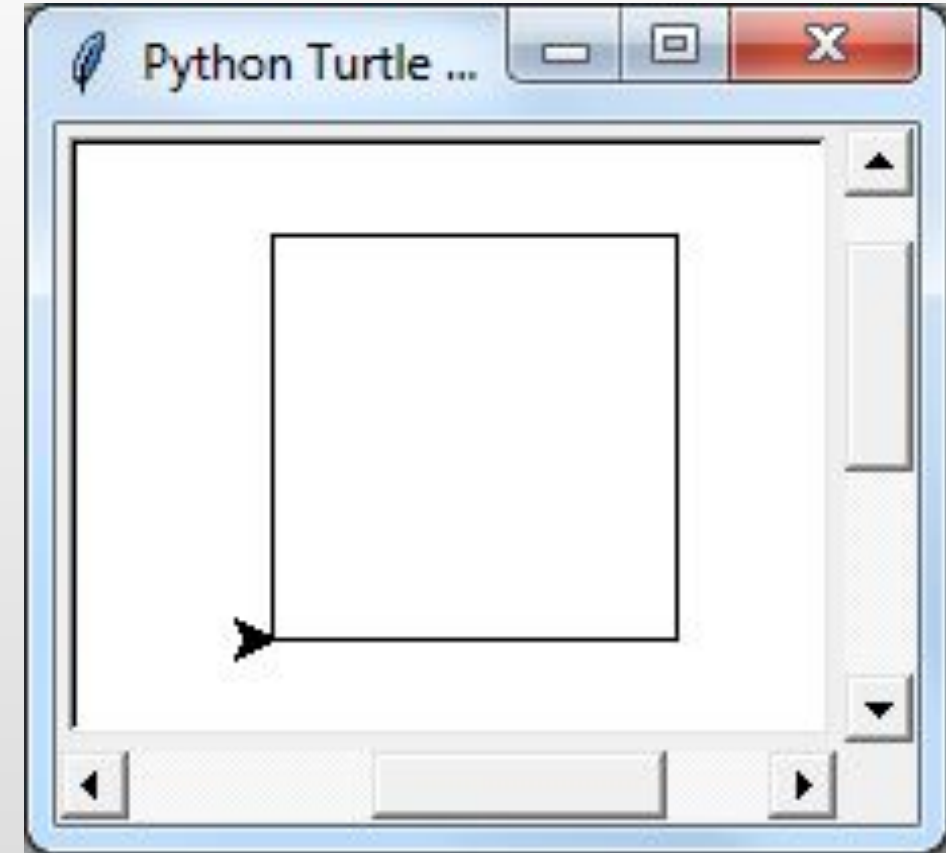
```
import turtle
timmy = turtle.Turtle()
timmy.forward(100)
timmy.left(90)
timmy.forward(100)
timmy.left(90)
timmy.forward(100)
timmy.left(90)
timmy.forward(100)
timmy.left(90)
```




How can we make this code shorter?



```
File Edit Format Run Options
import turtle
timmy = turtle.Turtle()
timmy.forward(100)
timmy.left(90)
timmy.forward(100)
timmy.left(90)
timmy.forward(100)
timmy.left(90)
timmy.forward(100)
timmy.left(90)
```





File Edit Format Run Options

```
import turtle
timmy = turtle.Turtle()
for a in range(4):
    timmy.forward(100)
    timmy.left(90)
```

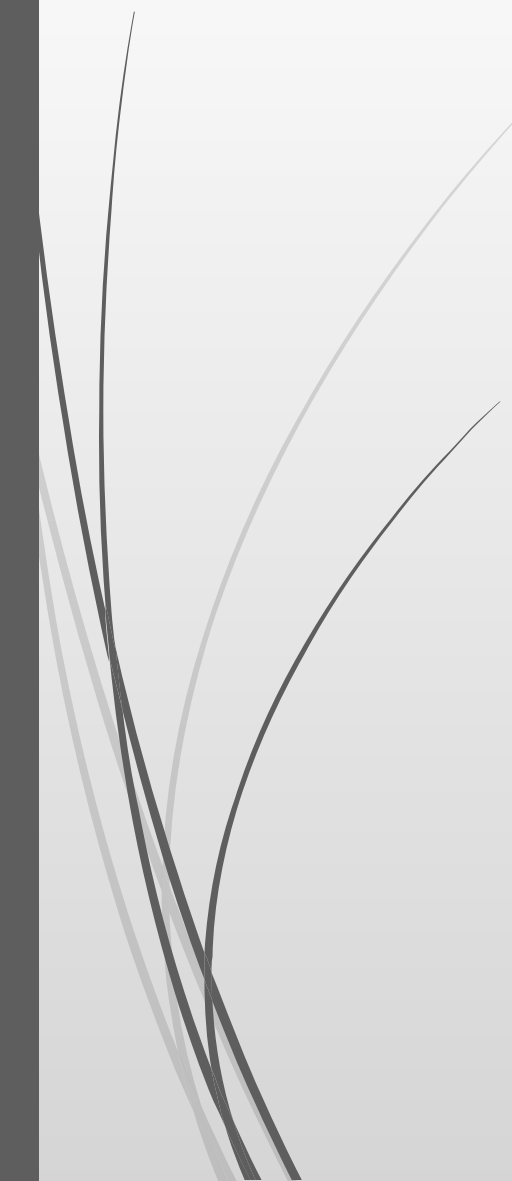


Moving your turtle

- ❑ You can also command your turtle to move to any coordinate on the screen by using the `turtle.goto()` function
 - ❑ This function accepts two arguments, `x` and `y`, denoting the `x` and `y` coordinates on the coordinate plane
- `turtle.goto(50,50)`
- ❑ Note however that your pen will continue to draw as you move your turtle to a new position



Moving your turtle

- ❑ You may have guessed that if your pen keeps on drawing, you may end up with a bunch of lines that you might not want ...
 - ❑ You can tell the turtle to stop drawing by using the `turtle.penup()` function. You can tell the turtle to start drawing again by using the `turtle.pendown()` function
- 



To Do:

Go to

<https://hourofpython.trinket.io/a-visual-introduction-to-python#/turtles/meet-tina> and

work through the exercises

Try the new exercises posted